EP 27241(5)

PCT

WORLD INTELLECTUAL PROPERTY ORGANIZATION
International Bureau

INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(51) **International Patent Classification** 6 : **G06F 9/44** — **A1**

(11) **International Publication Number:** **WO 99/63434**

(43) **International Publication Date:** 9 December 1999 (09.12.99)

(21) **International Application Number:** PCT/US99/08095

(22) **International Filing Date:** 13 April 1999 (13.04.99)

(30) **Priority Data:**
09/090,118 — 4 June 1998 (04.06.98) — US

(71) **Applicant:** GATEWAY 2000, INC. [US/US]; 610 Gateway Drive, P.O. Box 2000, North Sioux City, SD 57049-2000 (US).

(72) **Inventors:** KROENING, James, L.; 256 Firethorn Trail, Dakota Dunes, SD 57049 (US). FANGMAN, Darrin, J.; 8 Martha Street, McCook, SD 57049 (US). MARSHALL, James, J.; P.O. Box 332, Jefferson, SD 57038 (US). PEASLEY, Richard; 703 East Clay City, Elk Point, SD 57025 (US).

(74) **Agent:** VIKSNINS, Ann, S.; Schwegman, Lundberg, Woessner & Kluth, P.O. Box 2938, Minneapolis, MN 55402 (US).

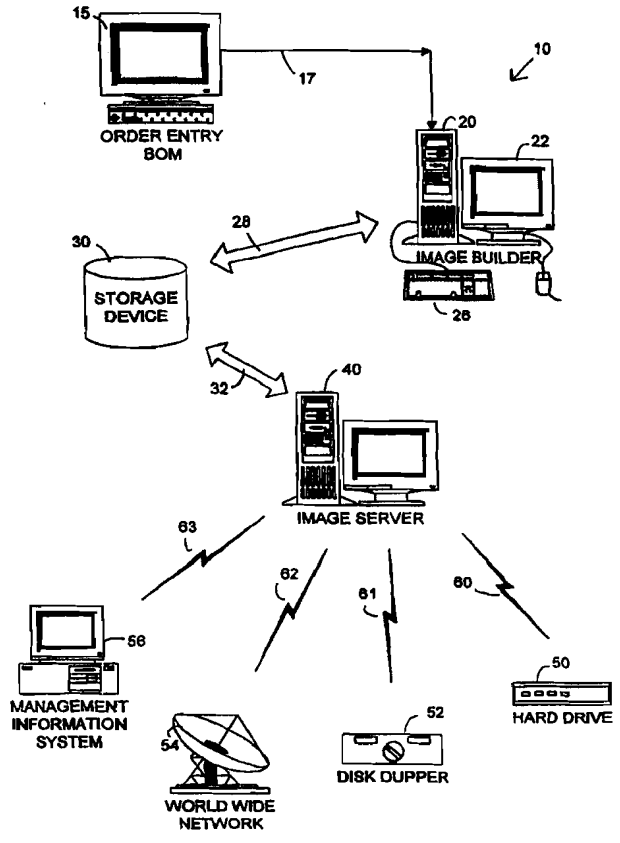
(81) **Designated States:** AU, CA, JP, European patent (AT, BE, CH, CY, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE).

Published
With international search report.
Before the expiration of the time limit for amending the claims and to be republished in the event of the receipt of amendments.

(54) **Title:** SOFTWARE DISTRIBUTION USING IMAGE BUILDER

(57) **Abstract**

A computerized system and a method for generating a custom software configuration for a hard drive of a computer system according to desired software configuration defined by a purchasing customer. The computerized system and method utilizes an image builder for creating a disk image of the desired software configuration and transferring the image to a storage device. The storage device is connected to an image server that transfers the disk image directly to the hard drive during a manufacturing and assembly process of the computer system. Once the computer system is assembled, the image server broadcasts the disk image to the computer system via a wired or wireless connection.



networked computers to have access to the new configuration. Networking prevents a system administrator from having to install software upgrades individually on each computer system. However, the upgrade must first be made to the file server which requires installation via a disk.

What is needed is a method or process for creating a desired software configuration without having to create a separate baseline configuration for each unique software configuration. This would save time and expense for a manufacturer or even a system administrator supporting a large number of computer work stations.

For the reasons stated above, and for other reasons stated below which will become apparent to those skilled in the art upon reading and understanding the present specification, there is a need in the art for a software creation/distribution process that is performed without having to physically set up a model system using a disk for each unique configuration desired.

Summary of the Invention

An image builder is a computerized network for generating a custom software configuration for a hard drive of a computer system according to a desired software configuration defined by a purchasing customer. The image builder surveys a database of software configurations to identify a match for a desired configuration or identify a baseline for the desired configuration. Once the baseline is identified, the image builder performs a comparison to create a set of changes that can be combined with or edited into the baseline image "picture" to generate the desired software configuration. This linear flow process utilizes intelligence and granularity in generating the desired software configuration.

In one embodiment, a computerized system for building a custom software configuration is provided. The computerized system comprises a processor, a computer readable medium, and a plurality of computer instructions executed by the processor from the computer readable medium for performing a series of steps. The steps include receiving a desired software configuration, surveying images of preexisting software configurations for a baseline software configuration similar to the desired software configuration, comparing an image of the baseline software configuration with the desired software configuration to determine a set of changes, and generating an image of the set of changes. The computerized system further includes the step of incorporating the image of the set of changes with the image of the baseline configuration to create the desired configuration.

In another embodiment, a computer readable medium having a computer readable program code embodied thereon is provided. The computer readable code comprising a computer readable code for causing a computer to receive a desired software configuration, a computer readable code for causing a computer to survey images of preexisting software configurations for a baseline software configuration similar to the desired software configuration, a computer readable code for causing a computer to compare an image of the baseline software configuration with the desired software configuration to determine a set of changes to be made to the baseline software configuration, and computer readable code for causing a computer to generate an image of the custom software configuration. The computer readable medium further comprises a computer readable program code for causing a computer to generate an image of the custom software configuration by incorporating the image of the set of changes with the image of the baseline configuration.

In yet another embodiment, a computerized network for creating and broadcasting a customized software configuration is provided. The computerized network comprises a computer system for receiving a desired software configuration, a storage device having a plurality of software configurations stored as images, and an image builder. The image builder receives the desired software configuration and surveys the storage device for an image of a baseline software configuration similar to the desired software configuration and compares the image with the desired software configuration to determine a set of changes, wherein the image builder generates an image of the custom software configuration by incorporating the image of the set of changes with the image of the baseline configuration and stores the image of the custom software configuration on the storage device. The computerized network also comprises an image server that retrieves the image of the custom software configuration from the storage device and broadcasts the image via a wired or wireless connection.

An advantage of the image builder is that common portions of multiple software configurations are identified by the image builder and are combined with a set of changes to produce the desired software configuration without having to reestablish a new baseline for each unique configuration. This means that baselines for different product lines, i.e., desk top computers, portable laptop computers, etc., can be combined with a set of changes to generate desired custom software configurations for various products produced by a manufacturer. Not having to create a separate baseline configuration for each possible configuration saves time and expense for a manufacturer or a system administrator supporting a large number of

work stations. Furthermore, since the image builder works on several levels of databases, it

The image builder 20 is coupled to the order entry system 15 via interface 17 for receiving the bill of materials which provides pertinent information for creating or building an image of the desired software configuration. Intelligence is provided in this step of the image delivery process because the image builder 20 sorts through a database of stored images to first determine if an image of the desired configuration has already been created for a prior computer configuration. These images are stored on a large capacity storage device 30 or multiple storage devices. If the image of the desired configuration has not previously been created, the image builder 20 selects an appropriate baseline image from the storage device 30 and then determines which incremental images are to be layered on top of the baseline image to achieve the desired final configuration.

These incremental images, which are also referred to as delta images, only contain additional information beyond the baseline image for achieving the desired software configuration. A level of granularity is thus achieved because of the linear flow process associated with achieving the desired configuration by adding a delta image to a baseline image. If the delta image is not in the data base, then the image builder constructs the appropriate delta image. Configuration numbers are assigned to all baseline images and delta images, which help the image builder 20 sort through all the possible images that can be used in the configuration process.

The actual steps that must be undertaken to create a digital image of a computing system's hard drive are well known to one skilled in the art. The image builder 20 looks at the data stream to be loaded on the computing system's hard drive 50. Since the disk image delivery process illustrated in Figure 1 is a linear process, the image builder 20 goes through the baseline image file by file and identifies those areas that are different and determines what parts are to be replaced. File names are examined not only by their name but by their creation date. The image builder 20 looks at specific files by a specific name or code at the time of its creation.

In lieu of examining the baseline on a file by file basis, a bit by bit comparison may be performed. In addition to the image builder 20 performing this comparison task, another level of intelligence is obtained when the image builder 20 determines changes to be made in registry settings and in interrupt settings so that the new software configuration will operate properly on the computing system. If the desired software configuration is not compatible with the hardware of the computing system, then the image builder rejects the bill of

Either the entire image (baseline and delta) or just the delta image may be loaded onto a computing system (not shown). Methods of loading either image onto a computing system are well known to those skilled in the art. If the computing system's hard drive, such as the one depicted in Figure 1, is currently configured and all that is needed is an application upgrade, then only the delta image is created and installed. However, if a new hard drive 50 is being configured, as in a manufacturing and assembly process, then the entire image is installed on the hard drive 50.

The image delivery process avoids having to create a new baseline for every image that is to be delivered. Furthermore, the image builder 20 contains editors that determine, for example, proper registry settings and also add directory information corresponding to installed applications and their file locations. An added benefit of layering information on top of a baseline is that it allows technical support personnel and software engineers to isolate problem areas and debug and correct problems as they arise. Because of the granularity, if a problem arises after a delta configuration was overlaid on top of baseline image, and that image is known to be error free, then it is easy to isolate the problem to the newly added delta configuration. Once another delta image is created with the fix, then the correct image is installed on the computing systems having the known error.

The image builder 20 is a computer having a processor (preferably, an Intel Pentium processor), random-access memory (RAM) (preferably, at least thirty-two megabytes), read-only memory (ROM), and one or more storage devices, such as a hard disk drive, a floppy disk drive (into which a floppy disk can be inserted), an optical disk drive, and a tape cartridge drive. The memory, hard drives, floppy disks, etc., are types of computer-readable media. The image builder 20 is coupled to a monitor 22, a pointing device 24 and a keyboard 26. The image builder is not limited to any particular type of computer. In one embodiment, the image builder 20 is a PC-compatible computer running a version of the Microsoft Windows operating system. The construction and operation of such computers are well known within the art.

The monitor 22 permits the display of information for viewing by a user of the computer. The invention is not limited to any particular monitor 22. Such monitors include cathode ray tube (CRT) displays, as well as flat panel displays such as liquid crystal displays (LCD's). The pointing device 24 permits the control of the screen pointer provided by the graphical user interface of operating systems such as versions of Microsoft Windows. The invention is not limited to any particular pointing device 24. Such pointing devices include

mouses, touch pads, trackballs, and point sticks. In one embodiment, the image builder 20 is a Gateway 2000, Inc., desktop personal computer, the monitor 22 includes a super-VGA CRT display, and the pointing device 24 is a mouse. Finally, the keyboard 26 permits entry of textual information into the image builder 20, as known within the art, and the image builder 20 is not limited to any particular type of keyboard.

The image builder 20 is coupled to a large volume storage device 30 via interface 28. Once the image builder 20 defines an image and assigns a corresponding configuration number, the images are stored on the storage device 30. Likewise, when the image builder 20 is creating a disk image, the storage device 30 is first surveyed to find the desired configuration or a close match to the desired configuration for establishing a baseline to build upon. The storage device 30 is not limited to any specific format or structure as long as the storage device 30 comprises a computer-readable media for interfacing. The storage device 30 may be a family of hard disk drives, a floppy disk drive (into which a floppy disk can be inserted), an optical disk drive, or a tape cartridge drive. The implemented storage device 30 may be internal to the image builder 20 or may exist as a stand lone entity, as illustrated in Figure 1.

After the image builder 20 has created an image of the desired software configuration, the image is passed from the storage device 30 to an image server 40 via interface 32. The image server 30 is another computer system similar to the image builder 20. The image server 30 is the point of delivery for the disk image. The image server 30 may have a variety of interfaces as illustrated in Figure 1. One illustrative embodiment has the image server coupled to a hard drive 50 via interface 60. This embodiment allows the hard drive 50 to be configured with an image of the desired software configuration before installation into a computer system. Another illustrative embodiment has the image server 30 coupled to a disk dupper 52 via interface 61. The disk dupper 52 duplicates the disk image on a computer readable medium, such as a floppy disk, a recordable CD, or a zip drive. Other means of copying the disk image are acceptable.

In still another illustrative embodiment, the image server 40 is coupled to a ground based transmitter 54 via interface 62 for wireless transmission of the disk image to an end user. Depending on the transmitter's 54 operating parameters and the transmitter's relay capabilities, the image could be transmitted anywhere in the world. In lieu of the ground based transmitter 54, the image server 40 is connected to the Internet in any particular manner, by which the invention is not limited to and which is not shown in Figure 1. Internet

connectivity is well known within the art. In one embodiment, the image server 40 includes a modem and corresponding communication drivers to connect to the Internet via what is known in the art as a "dial-up connection." In another embodiment, the image server 40 includes an Ethernet or similar hardware card to connect to a local-area network (LAN) that itself is connected to the Internet via what is known in the art as a "direct connection" (e.g., T1 line, etc.). In further embodiments, the image server 40 may be connected to the Internet using a cable modem or satellite Internet connectivity (as illustrated by transmitter 54).

In yet another illustrative embodiment, the image server 40 is coupled to a management information system (MIS) 56 via interface 63. Management Information Systems may be used to support the infrastructure of businesses and organizations wherein such systems are well known to one skilled in the art.

In referring to Figure 2 a logic flow for creating a disk image of a desired software configuration is illustrated. Block 200 is the start of the logic flow process, which represents receipt of the customer's order. Block 200 generates the Bill of Materials (BOM).

Receipt of the bill of materials is represented by block 204. The image builder 20 starts with the top record and calculates a configuration identification (ConFig ID) of all the entries. In block 208, the image builder 20 groups like orders together. Grouping like orders together allows for increased efficiency due to the commonality between orders.

In block 212, the image builder 20 compares the configuration IDs to the configuration history. If the configuration ID corresponds to a previously configured image, then the image builder 20 looks at whether the image is in a storage device 30, as illustrated in Figure 1. If the image is found in the storage device 30, then block 224 flags the configuration as ready for delivery and notifies an operator of the computerized network 10 that a desired image is ready. Otherwise, if the image is not found in the storage device 30, the image is created by the image builder 20 according to block 216 as a fresh build. As part of the fresh build process, block 230 requires the image builder 20 to process the bill of materials to determine the parameters for building an image according to the desired software configuration and ensure that they are compatible with the customer's hardware, software and special requirements. The final result or output from block 230 is an image or "digital picture" of the desired software configuration according to the bill of materials.

Figure 3 illustrates an architecture of a disk image 280 as created by the image builder 20. The image builder 20 builds the image 280 in software according to a desired software configuration and delivers that image to a storage device 30. Sections of the disk image 280

BNSDOCID: <WO_____9963434A1_I_>

are discussed in the order in which they are presented in Figure 3. One skilled in the art will readily realize other embodiments of an image architecture.

Section 300 contains BIOS flash properties. The next two sections, sections 302 and 304, contain the main operating system and the main applications' program code or instructions. Hardware characteristics of the computing system receiving the disk image 280 are addressed by section 306 dealing with the CMOS settings, section 308 includes the main BIOS instructions, section 310 supports LOC information and section 312 supports desktop parameters for the main operating system.

Section 314 supports information including, but not limited to the following: operating system registers, initialization information and configurations files. Section 316 includes test information. Similar to section 314, section 318 includes, but is not limited to the following: application system registers, initialization information and configurations files. The last two sections contain an identification of the specific image itself 320, and the last section contains an identification of the customer 322. The identification numbers allow for future reference of the created image, which is helpful for trouble shooting problems in the software configuration and in also adding delta images to the previously delivered image in order to upgrade existing applications.

Figure 4 illustrates an exemplary embodiment of an identification scheme 380 for a disk image structure. The image identification is a tree structure with a configuration identification number. Also included in the tree structure are underlying identification numbers corresponding to main files and underlying identification numbers corresponding to edited dynamic files (EDF).

More specifically, the identification scheme 380 includes a configuration number 380 identifying what the desired image is built from. It is the foundation from which the image builder 20 works from in creating the desired image. Once the configuration ID 380 is identified, then the main files corresponding to the operating system 402, e.g., Windows 95, and the desired application 404 are layered on top of the configuration ID 400 basic files. Edited dynamic files corresponding to registry settings 406, operating system initialization files 408, application EDF files 410.

An image build software delivery process has been described. The process creates an image of a customer's order in software before placing the image on a hard drive or other storage means for the customer. Once an image has been created, changes or deltas to the baseline image can easily be made without having to redefine the baseline. Adding a delta

image to the baseline image allows the desired image to be achieved. This method provides levels of granularity wherein incremental changes can be made to a system without having to perform major work by redefining the baseline. This allows for easy upgrades and allows technical support to function efficiently.

Although specific embodiments have been illustrated and described herein, it will be appreciated by those skilled in the art that any arrangement which is calculated to achieve the same purpose may be substituted for the specific embodiment of the present invention. Therefore, it is manifestly intended that this invention be limited only by the following claims and equivalents.

What is claimed is:

1. A method of building a custom software configuration, the method comprising the steps of:

receiving a desired software configuration;

surveying images of preexisting software configurations for a baseline software configuration;

comparing an image of the baseline software configuration with the desired software configuration; and

generating an image of a set of changes based on the comparison.

2. The computerized system of claim 1 wherein the plurality of computer instructions further performs the step of broadcasting the image of the set of changes.

3. The method of claim 1 further comprises the step of building an image of the custom software configuration by incorporating the image of the set of changes with the image of the baseline configuration.

4. The computerized system of claim 3 wherein the plurality of computer instructions further performs the step of broadcasting the image of the custom software configuration.

5. The method of claim 4 wherein the step of broadcasting is performed by a satellite.

6. The method of claims 4 wherein the step of broadcasting is performed over the Internet.

7. The method of claims 4 wherein the step of broadcasting is performed using an Ethernet connection for connecting to a local area network.

8. The method of claim 1 wherein the step of comparing the image of the baseline software configuration with the desired software configuration is performed by examining a data stream of the baseline configuration on a bit by bit basis.

9. The method of claim 1 wherein the step of comparing the image of the baseline software configuration with the desired software configuration is performed by examining a data stream of the baseline configuration on a file by file basis.

10. The method of claim 1 further comprises the step of storing the image of the set of changes on a computer readable medium.

11. A computerized system for building a custom software configuration, the computerized system comprising:

a processor;

a computer readable medium; and

a plurality of computer instructions executed by the processor from the computer readable medium for performing the steps of:

receiving a desired software configuration,

surveying images of preexisting software configurations for a baseline software configuration similar to the desired software configuration,

comparing an image of the baseline software configuration with the desired software configuration to determine a set of changes; and

generating an image of the set of changes.

12. The computerized system of claim 11 wherein the plurality of computer instructions further performs the step of broadcasting the image of the set of changes.

13. The computerized system of claim 11 wherein the plurality of computer instructions further comprises the step of building an image of the custom software configuration by incorporating the image of the set of changes with the image of the baseline configuration.

14. The computerized system of claim 13 wherein the plurality of computer instructions further performs the step of broadcasting the image of the custom software configuration.

15. The computerized system of claim 14 wherein the step of broadcasting is performed by a satellite.

BNSDOCID: <WO_____9963434A1_I_>

16. The computerized system of claim 14 wherein the step of broadcasting is performed over the Internet.

17. The computerized system of claim 14 wherein the step of broadcasting is performed using the Ethernet connection for connecting to a local area network.

18. The computerized system of claim 11 wherein the step of comparing the image of the baseline software configuration with the desired software configuration is performed by examining a data stream of the baseline configuration on a file by file basis.

19. The computerized system of claim 11 wherein the step of comparing the image of the baseline software configuration with the desired software configuration is performed by examining a data stream of the baseline configuration on a bit by bit basis.

20. The computerized system of claim 11 wherein the plurality of computer instructions further performs the step of storing the image of the set of changes on the computer readable medium.

21. A machine readable medium having machine readable instructions stored thereon for causing a computer to perform the steps comprising:

receiving a desired software configuration;

surveying images of preexisting software configurations for a baseline software configuration similar to the desired software configuration;

comparing an image of the baseline software configuration with the desired software configuration to determine a set of changes to be made to the baseline software configuration; and

generating an image of the custom software configuration.

22. The machine readable medium of claim 21 further comprises the step of generating an image of the custom software configuration by incorporating the image of the set of changes with the image of the baseline configuration.

BNSDOCID. <WO_____9963434A1_I_>

23. The machine readable medium of claim 21 further comprises the step of broadcasting the set of changes.

24. The machine readable medium of claim 21 further comprises the step of storing the image of the set of changes on a computer readable medium.

25. A computerized network for creating and broadcasting a customized software configuration, the computerized network comprising:

a computer system for receiving a desired software configuration;

a storage device having a plurality of software configurations stored as images;

an image builder that receives the desired software configuration and surveys the storage device for an image of a baseline software configuration similar to the desired software configuration and compares the image with the desired software configuration to determine a set of changes, wherein the image builder generates an image of the custom software configuration by incorporating the image of the set of changes with the image of the baseline configuration and stores the image of the custom software configuration on the storage device; and

an image server that retrieves the image of the custom software configuration from the storage device and broadcasts the image.

26. The computerized network of claim 25 wherein the image server broadcasts the image via a wireless connection.

27. The computerized network of claim 25 wherein the image server broadcasts the image via a wired connection.

28. A method of updating a software configuration of a computer system, the method comprising the steps of:

a means for receiving a desired software configuration update;

a means for examining an image of a baseline software configuration corresponding to the software configuration;

a means for comparing the baseline software configuration with the desired software configuration update to determine a set of changes; and

a means for generating an image of the set of changes that creates the desired software configuration update when combined with the image of the baseline configuration.

29. The method of updating a software configuration of claim 28 wherein the means for generating an image of the set of changes further include a means for broadcasting the image of the set of changes.

30. The computerized network of claim 29 wherein the means for broadcasting is performed via a wireless connection.

31. The computerized network of claim 29 wherein the means for broadcasting is performed via a wired connection.

32. A computerized system for building a custom software configuration, the computerized system comprising:

a processor;

a storage device accessible to the processor having a plurality of software configurations stored as images; and

an image builder that receives the desired software configuration and surveys the storage device for an image of a baseline software configuration similar to the desired software configuration and compares the image with the desired software configuration to determine a set of changes, wherein the image builder generates an image of the custom software configuration by incorporating the image of the set of changes with the image of the baseline configuration and stores the image of the custom software configuration on the storage device

BNSDOCID: <WO_____9963434A1_I_>

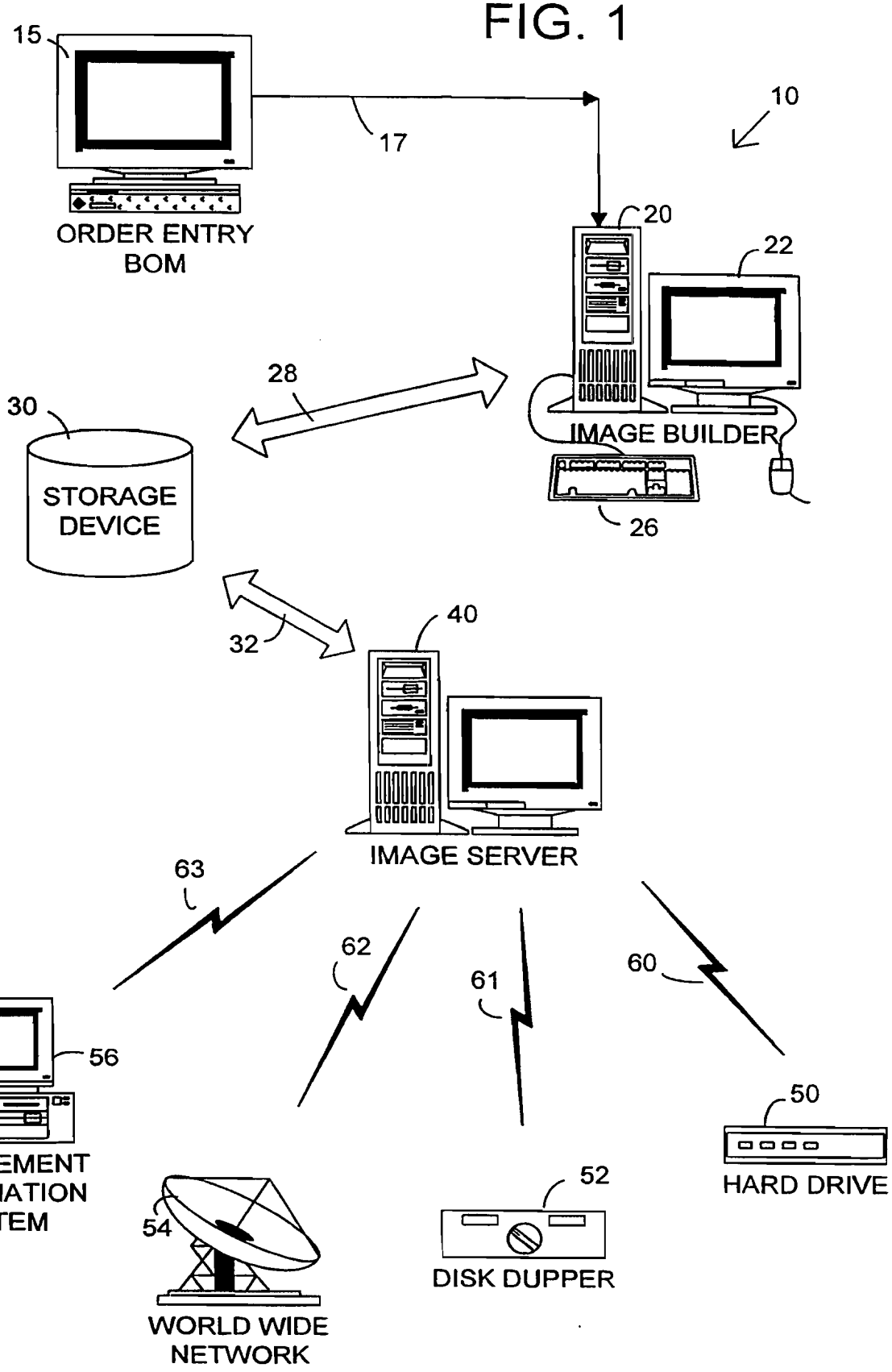
33. The computerized system of claim 32 further comprising an image server for broadcasting the image.

34. The computerized system of claim 33 wherein the image is broadcast via a satellite.

35. The computerized system of claim 33 wherein the image is broadcast over the Internet.

36. The computerized system of claim 33 wherein the image is broadcast using an Ethernet connection connected to a local area network.

FIG. 1



BNSDOCID: <WO_____9963434A1_I_>

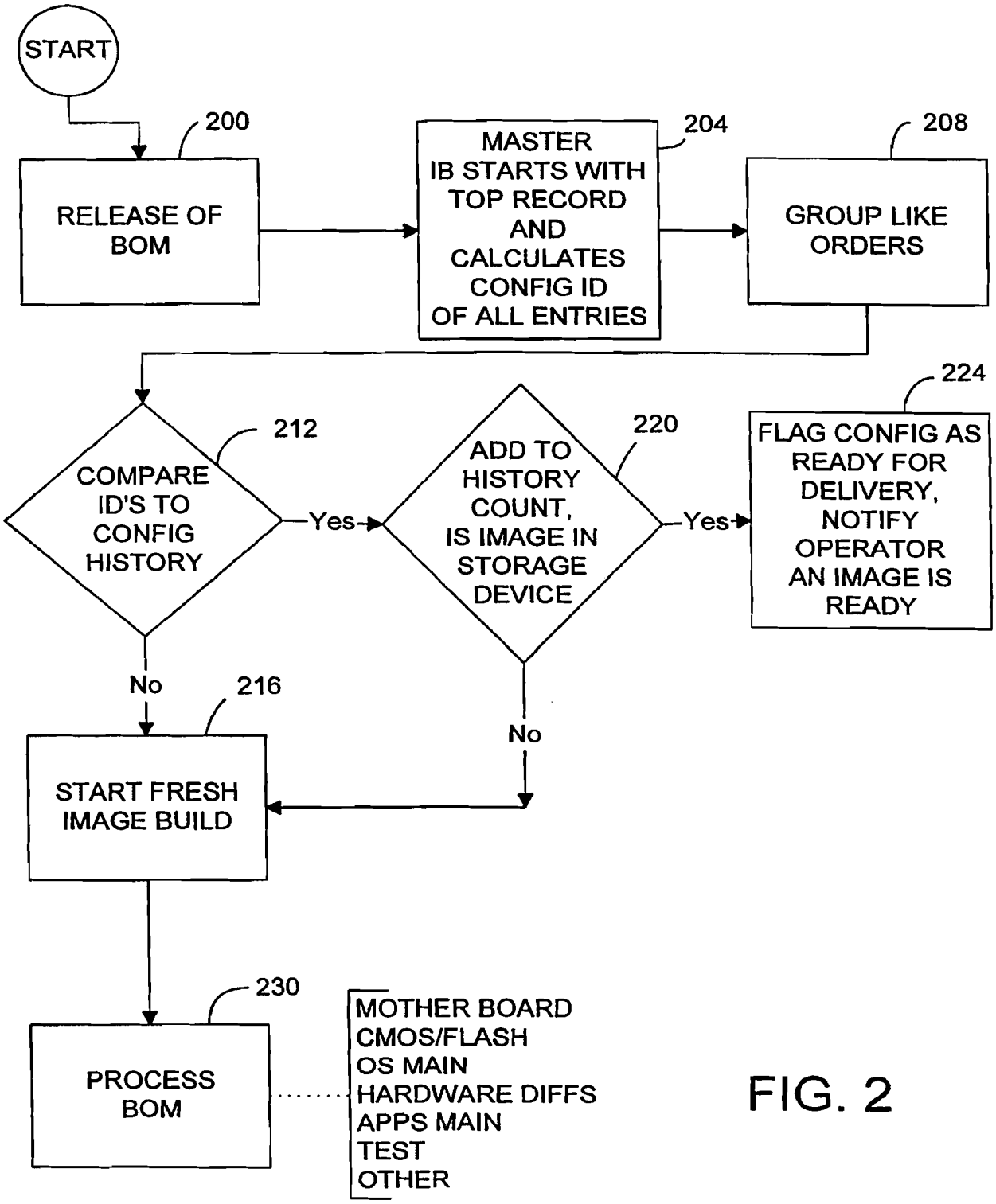

START
200
RELEASE OF BOM
204
MASTER IB STARTS WITH TOP RECORD AND CALCULATES CONFIG ID OF ALL ENTRIES
208
GROUP LIKE ORDERS
224
FLAG CONFIG AS READY FOR DELIVERY, NOTIFY OPERATOR AN IMAGE IS READY
212
COMPARE ID'S TO CONFIG HISTORY
Yes
220
ADD TO HISTORY COUNT, IS IMAGE IN STORAGE DEVICE
Yes
No
216
No
START FRESH IMAGE BUILD
230
PROCESS BOM
MOTHER BOARD
CMOS/FLASH
OS MAIN
HARDWARE DIFFS
APPS MAIN
TEST
OTHER

FIG. 2

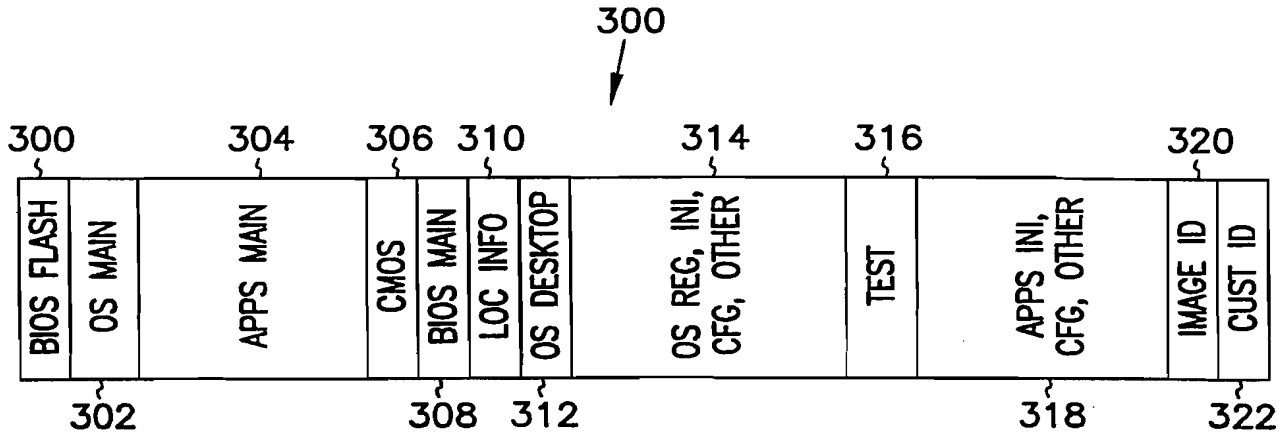


FIG. 3

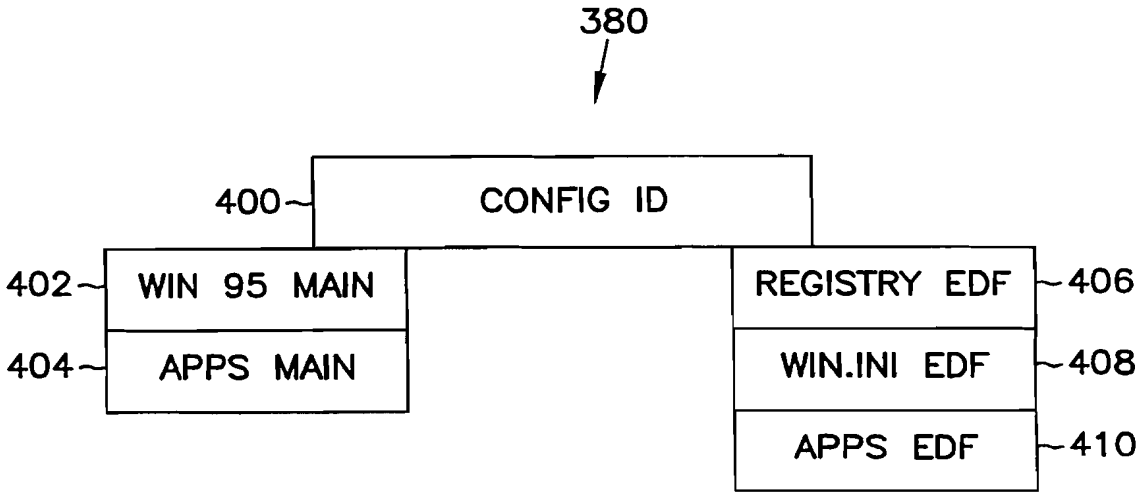


FIG. 4

BNSDOCID: <WO_____9963434A1_I_>

INTERNATIONAL SEARCH REPORT

Interi nal Application No
PCT/US 99/08095

A. CLASSIFICATION OF SUBJECT MATTER
IPC 6 G06F9/44

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)
IPC 6 G06F

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practical, search terms used)

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category °	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	EP 0 598 505 A (CANON INFORMATION SYST INC) 25 May 1994 (1994-05-25) abstract page 2, line 36 -page 3, line 7 page 16, line 19 -page 17, line 23 claim 1	1,4-7, 11, 14-17, 21, 25-28, 30-36
A	DE 21 63 281 A (IBM) 7 December 1972 (1972-12-07) page 4, line 1 -page 5, line 17 page 8, line 26 -page 11, line 14; figure 2	1,11,21, 25,28,32

-/--

[X] Further documents are listed in the continuation of box C.

[X] Patent family members are listed in annex.

° Special categories of cited documents :

"A" document defining the general state of the art which is not considered to be of particular relevance

"E" earlier document but published on or after the international filing date

"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art.

"&" document member of the same patent family

Date of the actual completion of the international search	Date of mailing of the international search report
19 October 1999	28/10/1999

Name and mailing address of the ISA	Authorized officer
European Patent Office, P.B. 5818 Patentlaan 2 NL – 2280 HV Rijswijk Tel. (+31–70) 340–2040, Tx. 31 651 epo nl, Fax: (+31–70) 340–3016	Bijn, K

1

Form PCT/ISA/210 (second sheet) (July 1992)

BNSDOCID: <WO_____9963434A1_I_>

INTERNATIONAL SEARCH REPORT

International Application No
PCT/US 99/08095

C.(Continuation) DOCUMENTS CONSIDERED TO BE RELEVANT

Category °	Citation of document, with indication,where appropriate, of the relevant passages	Relevant to claim No.
A	EP 0 717 353 A (AT & T CORP) 19 June 1996 (1996-06-19) abstract page 10, line 4 -page 12, last line	1-36
A	US 5 604 906 A (MURPHY COLM J ET AL) 18 February 1997 (1997-02-18) column 6, line 10 -column 9, line 53	1,4-7, 10,11, 14-17, 20,21, 24-28, 30-36
A	EP 0 528 617 A (SUN MICROSYSTEMS INC) 24 February 1993 (1993-02-24) column 4, line 33 -column 5, line 7 column 7, line 47 -column 10, line 33	1-3,5-7, 10-13, 15-17, 21-23, 25-36

1

Form PCT/ISA/210 (continuation of second sheet) (July 1992)

BNSDOCID: <WO_____9963434A1_I_>

INTERNATIONAL SEARCH REPORT

...information on patent family members

Interr ...al Application No
PCT/US 99/08095

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
EP 0598505 A	25-05-1994	US 5530862 A	25-06-1996
		JP 6214910 A	05-08-1994
DE 2163281 A	07-12-1972	GB 1327962 A	22-08-1973
		IT 1006009 B	30-09-1976
EP 0717353 A	19-06-1996	BR 9505888 A	06-01-1998
		CA 2161360 A	15-06-1996
		CN 1132372 A	02-10-1996
		JP 8255104 A	01-10-1996
		SG 34329 A	06-12-1996
US 5604906 A	18-02-1997	NONE	
EP 0528617 A	24-02-1993	JP 7044373 A	14-02-1995
		US 5481722 A	02-01-1996

BNSDOCID: <WO_____9963434A1_I_>